ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 2, February 2018

High Performance Packet Capturing Using Data Plane Development Kit (DPDK)

Sayali Divekar, Hardika Shah, Niranjan Kulkarni, Sonali Argade

Warje, Pune, Maharashtra, India

ABSTRACT: High packet loss, more time required to capture the packets etc. are the major problems in wireless communication network. In data plane, packets are transferred to kernel and from there to user space. Using tools like Data Plane Development Kit (DPDK), this problem can be greatly reduced. High packet capturing tool (PackCap) is proposed by porting TCPdump on DPDK, so that packets can be captured at the rate of 1 GBps. The kernel is bypassed in this PackCap tool and only data plane packets are considered. Network administrator can inspect packets to diagnose, solve network problems and to determine whether network security policies are being followed.

General Terms
PackCap, packet capturing, Packet processing,.

KEYWORDS: DPDK, NFV, EAL, TCPDUMP.

I. INTRODUCTION

Packet capturing tools are used greatly now-a-days to know how the packets are traversed in the network. Recent advances of server-virtualization technology and cloud computing technology allows many services, such as web, database, and application, run with virtualized computing resources in a data center.[7] Virtual machines need to communicate and access peripherals, which for systems used as servers mostly means disks and network interfaces.[5] The network interface (NIC) is normally able to manage circular lists (called NIC rings) of memory buffers, and move packets between the physical links and these buffers without CPU intervention.[6]. Packet processing at a switch consists of: (a) packet reception, (b) packet header parsing, (c) packet classification, (d) flow-table lookup to determine outgoing port, (e) QoS mechanisms such as policing on the ingress and scheduling on the egress, and (f) packet transmission [10].

II. RELATED WORK

In the 2008 Nick McKeown, et al [1] proposed the Run experiments on heterogeneous switches and no need for vendors to expose but reuse of controllers is avoided.
In the year 2009 Steen Larsen, et al [2] proposed End to end latency and Adaptive interrupt moderation but Variability and I/O performance on virtual machine is limited.

In 2010 Binbin Zhang, et al [3] developed I/O performance and Optimization of Kernel Virtual Machine but guest OS is simple.
In the year 2010 Sangjin Han, et al [4] proposed that the system Outperforms software routers by more than a factor of four and Minimized per-packet processing overhead but the system is not available publically.
In 2012 L. Rizzo, et al [5] developed the system with highspeed communication but not compactible with better hypervisor.
In the year 2014 Y. Nakajima, et al [7] proposed that Use of DPDK accelerates network I/O performance by bypassing packet processing in OS kernel and direct access to NIC packet buffer from data-plane program but In case of long packet, the switch perform 10-Gbps wirerate.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 2, February 2018

In 2015 Michio Honda , et al [9] stated that A flexible software switch providing high throughput, low CPU utilization and which can switch packets at rates of hundreds of gigabits per second but it Receives per packet semantics instead of batches of packet decreases the performance.
In the year 2015 Tom Barbette, et al [10] proposed that DPDK FastClick implementations removes the important overhead for small packets, mostly by removing the Click wiring cost by using batches and reducing the cost of the packet pool, using I/O batching and FastClick improved abstraction in packet processing but The flow capabilities of fastClick is not good for the development of Middleboxes.
In 2015 Sebastian Gallenmüller, et al [11] stated that DPDK seems to be superior to netmap. It uses well known OS interfaces and modified system calls for packet IO, leading to increased performance but The high data rate can be achieved by modifying the interfaces he limitation is that this system requires appplication to be ported to one of the frameworks.
In the year 2016 Reza Rahimi, et al [14] proposed that loop-count variable is used to control packet batching. Packet drop rate could be non-zero when the OpenFlow table size could be large but there is need for careful calibration andplanning parallelization and a software switch should be calibrated for the highest packet arrival rate that it can handle with zero packet drops.

III. SYSTEM ARCHITECTURE

The proposed system describes the architecture of the Data Plane Developement Kit. The DPDK is the frame work which contains different set of libraries. DPDK is used for packet processing. DPDK was first supported by X86. It runs on Linux operating system. DPDK is open source framework. Main libraries of DPDK are ring buffers huge page libraries, poll driver modes. These libraries are used for sending and receiving packets in less number of CPU cycles. The system architecture fig 1.1 proposed helps in fast packet processing from network layer to application layer.
The DPDK includes different set of data plane libraries. Libraries of DPDK are created with the help of Environment Abstraction Layer. EAL hides the specific requirement of environment and provides standard programs for interface.
The system contains Network Interface Controller which can also be called as Network interface card or Network adapter, Lan adapter , physical network interface which connects computer to computer network. The Environment Abstraction Layer is used to gain acesss to hardware and memory space. They are used for assigning execution units to specific cores. EAL is also used for creating instances.
EAL is used to reserve memory zones. It is used to provide interface to access PCI address space. Functions like tracing and debugging are included in EAL. Packets are transfered from NIC to Poll mode driver library. Packets bypasses the kernel space and directly transfers packets to userspace. Kernelspace contains EAL. In DPDK the packets are directly transfered from NIC to userspace excluding kernel space.
Poll mode driver library has application programming interface provided by BSD driver which runs on userspace, for configuring devices and their queues. DPDK has 1GB, 10GB,
40GB with para virtualised virtio poll mode drivers. Packet is transferred to DPDK pool and DPDK ring.
Packets are transfered to DPDK pool and packet description to DPDK ring. Ring is the library which is used for management of queues. Maximum size is fixed and pointers are stored in ring library of DPDK. Finally packets are transfered to application layer.

IV. CONCLUSION

In this paper, fast CPU is important for efficient and predictable performance. DPDK is best performance predictable approach for packet forwarding in a network.SR-IOV should be avoided in production networks because of the performance characteristics. Packet forwarding performance of NFV-nodes is under the attention especially in the fields of carrier and datacenter networks where the concept of NFV has already been deployed. NFV brings in lots of benefits in network and service management, but low and unstable performance nature of NFV-nodes prevents full-scale deployment of NFV. Countless performance improvement methods have been proposed so far, and network engineers and administrators have various options for NFV-node system.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 2, February 2018

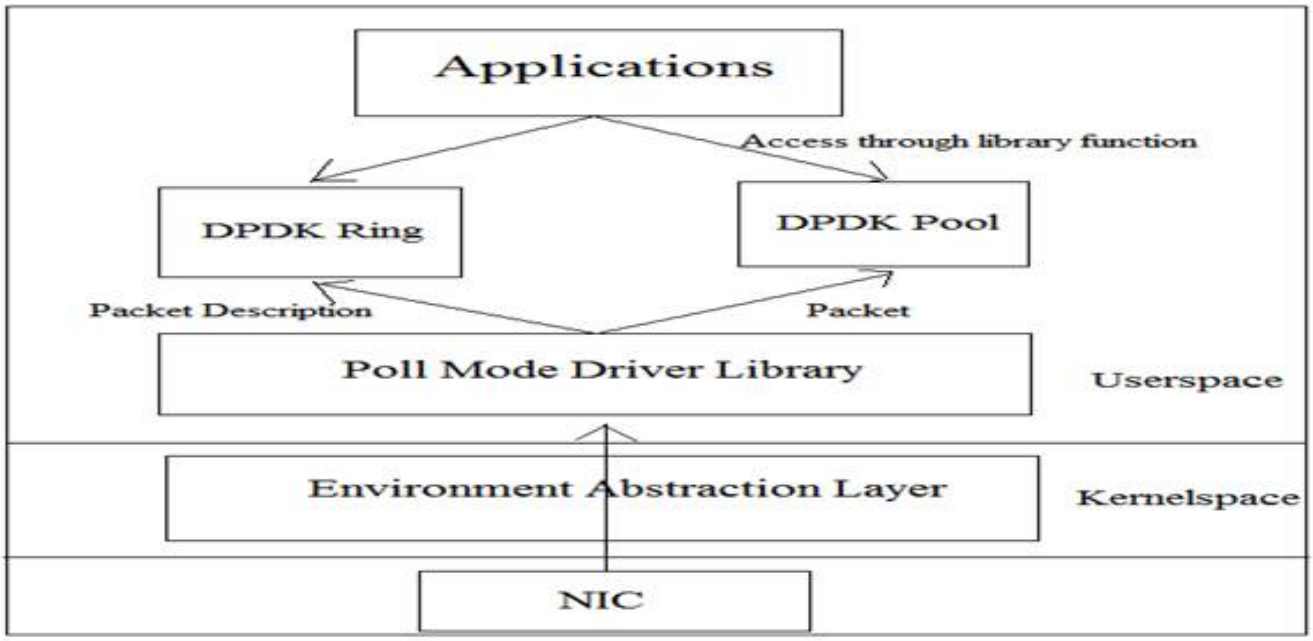

Fig 1: System Architecture

V. ACKNOWLEDGMENTS

It gives us great pleasure in presenting the preliminary project report on 'High Speed Packet Capturing Using Data Plane Development Kit (DPDK)'.
We would like to take this opportunity to thank my internal guide Prof. Sonal Fatangare for giving me all the help and guidance I needed. I am really grateful to them for their kind support. Their valuable suggestions were very helpful.
We are also grateful to Prof. Vina Lomte, Head of Computer Engineering Department, RMD Sinhgad School Of Engineering for her indispensable support, suggestions.

REFERENCES

[1] Nick McKeown, Tom Anderson, Hari Balakrishnan, Guru Parulkar, Larry Peterson, Jennifer Rexford, Scott Shenker, Jonathan Turner, "OpenFlow: Enabling Innovation in Campus Networks", ACM SIGCOMM Computer Communication Review, vol. 38, no. 2, pp. 69-74, April 2008
[2] Steen Larsen, Parthasarathy Sarangam, Ram Huggahalli, "Architectural Breakdown of End-to-End Latency in a TCP/IP Network", IEEE, 2009.
[3] Binbin Zhang, Xiaolin Wang, Rongfeng Lai, Liang Yang, Zhenlin Wang,Yingwei Luo, Xiaoming Li, "Evaluating and Optimizing IOVirtualization in Kernel-based Virtual Machine (KVM)", IFIInternational Conference on Network and Parallel Computing (NPC), pp. 220–231, 2010.
[4] Sangjin Han, Keon Jang, KyoungSoo Park, Sue Moon, "PacketShader: a GPU-Accelerated Software Router", ACM SIGCOMM 2010, pp. 195– 206, Delhi, India, Sep. 2010.
[5] L. Rizzo and G. Lettieri, "VALE, a Switched Ethernet for. Virtual Machines", 8th International Conference on emerging Networking Experiments and Technologies (CoNEXT'12), pp. 61–72, 2012..
[6] L. Rizzo, M. Carbone, and G. Gatalli, "Transparent acceleration of software packet forwarding using netmap", IEEE INFOCOM, pp. 2471–2479, 2012.
[7] Y. Nakajima, T. Hibi, H. Takahashi, H. Masutani, K. Shimano, and M. Fukui, "Scalable, High-performance, Elastic Software OpenFlow Switch in Userspace for Wide-area Network", Open Networking Summit (ONS 2014), Santa Clara, CA, March 2014..
[8] R. Bonafiglia, I. Cerrato, F. Ciaccia, M. Nemirovsky, F. Risso, "Assessing the Performance of Virtualization Technologies for NFV: a Preliminary Benchmarking.", Fourth European Workshop on Software Defined Networks (EWSDN), pp. 67–72, Bilbao, Spain, 2015.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 2, February 2018

[9] Michio Honda , Felipe Huici, Giuseppe Lettieri, Luigi Rizzo, "mSwitch: A Highly-Scalable, Modular Software Switch", ACM SIGCOMM Sym- posium on Software Defined Networking Research (SOSR), June 2015.

[10] Tom Barbette, Cyril Soldani, Laurent Mathy, "Fast Userspace Packet Processing", ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS), pp. 5–16, Oakland, CA, May 2015.

[11] Sebastian Gallenmuller, Paul Emmerich, Florian Wohlfart, Daniel Raumer, and Georg Carle, "Comparison of Frameworks for High-Performance Packet IO", ACM/IEEE Symposium on Architectures for Networking and Commu- nications Systems (ANCS), pp. 29–38, Oakland, CA, May 2015.

[12] Paul Emmerich, Sebastian Gallenmuller, Daniel Raumer, Florian Wohlfart, and Georg Carle, "MoonGen: A Scriptable High-Speed Packet Generator", ACM Conference on Internet Measurement Conference (IMC), pp. 275–287, 2015.

[13] Michail-Alexandros Kourtis,Gorgios Xilouris,Vincenzo Riccobene,Michael J.Mcgarth, Giuseppe Petralia, "Enhancing VNF performance by exploiting SR –IOV and DPDK Packet Processing Acceleration", IEEE Conference on Network Function Virtualization and Software Defined Network (NFV-SDN), pp. 74–78, San Francisco Nov. 2015.

[14] Reza Rahimi & M. Veeraraghavan ,Y. Nakajima & H. Takahashi Y. Nakajima, S. Okamoto & N. Yamanaka, "A High-Performance OpenFlow Software Switch", IEEE 17th International Conference on High Performance Switching and Routing (HPSR), pp. 93–99, June 2016.

[15] Ryota Kawashima, Tsunemasa Hayashi, Hiroshi Matsuo, "Evaluation of Forwarding Efficiency in NFV-nodes toward Predictable Service Chain Performance", IEEE transaction of network services and management, 2017.